JDBC - ENVIRONMENT SETUP

To start developing with JDBC setup your JDBC environment by following the steps shown below. We assume that you are working on a Windows platform.

Install Java:

Install J2SE Development Kit 5.0 (JDK 5.0) from Java Official Site.

Make sure following environment variables are set as described below:

- **JAVA_HOME:** This environment variable should point to the directory where you installed the JDK, e.g. C:\Program Files\Java\jdk1.5.0

- **CLASSPATH:** This environment variable should has appropriate paths set, e.g. C:\Program Files\Java\jdk1.5.0_20\jre\lib

- **PATH:** This environment variable should point to appropriate JRE bin, e.g. C:\Program Files\Java\jre1.5.0_20\bin.

It is possible you have these variable set already, but just to make sure here's how to check.

- Go to the control panel and double-click on System. If you are a Windows XP user it's possible you have to open Performance and Maintenance before you will see the System icon.

- Go to the Advanced tab and click on Environment Variables.

- Now check all the above mentioned variables are set properly.

You automatically get both JDBC packages **java.sql** and **javax.sql** when you install J2SE Development Kit 5.0 (JDK 5.0)

Install Database:

The most important thing you will need, of course is an actual running database with a table that you can query and modify.

Install a database that is most suitable for you. You can have plenty of choices and most common are:

- **MySQL DB:** MySQL is an open source database. You can download it from MySQL Official Site. We recommend downloading the full Windows installation.

  In addition, download and install MySQL Administrator as well as MySQL Query Browser. These are GUI based tools that will make your development much easier.

  Finally, download and unzip MySQL Connector/J (the MySQL JDBC driver) in a convenient directory. For the purpose of this tutorial we will assume that you have installed the driver at C:\Program Files\MySQL\mysql-connector-java-5.1.8.

  Accordingly set CLASSPATH variable to C:\Program Files\MySQL\mysql-connector-java-5.1.8\mysql-connector-java-5.1.8-bin.jar. Your driver version may vary based on your installation.

- **PostgreSQL DB:** PostgreSQL is an open source database. You can download it from PostgreSQL Official Site.

The Postgres installation contains a GUI based administrative tool called pgAdmin III. JDBC drivers are also included as part of the installation.

- **Oracle DB:** Oracle DB is an commercial database sold by Oracle . We assume that you have the necessary distribution media to install it.

   Oracle installation includes a GUI based administrative tool called Enterprise Manager. JDBC drivers are also included as part of the installation.

Install Database Drivers:

The latest JDK includes a JDBC-ODBC Bridge driver that makes most Open Database Connectivity (ODBC) drivers available to programmers using the JDBC API.

Now a days most of the Database vendors are supplying appropriate JDBC drivers along with Database installation. So you should not worry about this part.

Set Database Credential:

For this tutorial we are going to use MySQL database. When you install any of the above database, its administrator ID is set to **root** and gives provision to set a password of your choice.

Using root ID and password you can either create another users ID and password or you can use root ID and password for your JDBC application.

There are various database operations like database creation and deletion, which would need administrator ID and password.

For rest of the JDBC tutorial we would use MySQL Database with **username** as ID and **password** as password.

If you do not have sufficient privilege to create new users then you can ask your Database Administrator (DBA) to create a user ID and password for you.

Create Database:

To create the **EMP** database, use the following steps:

Step 1:

Open a **Command Prompt** and change to the installation directory as follows:

```
C:\>
C:\>cd Program Files\MySQL\bin
C:\Program Files\MySQL\bin>
```

Note: The path to **mysqld.exe** may vary depending on the install location of MySQL on your system. You can also check documentation on how to start and stop your database server.

Step 2:

Start the database server by executing the following command, if it is already not running.

```
C:\Program Files\MySQL\bin>mysqld
C:\Program Files\MySQL\bin>
```

Step 3:

Create the **EMP** database by executing the following command

```
C:\Program Files\MySQL\bin> mysqladmin create EMP -u root -p
Enter password: ********
C:\Program Files\MySQL\bin>
```

Create Table

To create the **Employees** table in EMP database, use the following steps:

Step 1:

Open a **Command Prompt** and change to the installation directory as follows:

```
C:\>
C:\>cd Program Files\MySQL\bin
C:\Program Files\MySQL\bin>
```

Step 2:

Login to database as follows

```
C:\Program Files\MySQL\bin>mysql -u root -p
Enter password: ********
mysql>
```

Step 3:

Create the table **Employee** as follows:

```
mysql> use EMP;
mysql> create table Employees
    -> (
    -> id int not null,
    -> age int not null,
    -> first varchar (255),
    -> last varchar (255)
    -> );
Query OK, 0 rows affected (0.08 sec)
mysql>
```

Create Data Records

Finally you create few records in Employee table as follows:

```
mysql> INSERT INTO Employees VALUES (100, 18, 'Zara', 'Ali');
Query OK, 1 row affected (0.05 sec)

mysql> INSERT INTO Employees VALUES (101, 25, 'Mahnaz', 'Fatma');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO Employees VALUES (102, 30, 'Zaid', 'Khan');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO Employees VALUES (103, 28, 'Sumit', 'Mittal');
Query OK, 1 row affected (0.00 sec)

mysql>
```

For a complete understanding on MySQL database, study MySQL Tutorial.

Now you are ready to start experimenting with JDBC. Next tutorial would give your a sample example on JDBC Programming.